

Using the Super Natural RPC Service

This section describes how to use the RPC Service.

This section covers the following topics:

- Introduction
 - Hints and Limitations
 - Dynamic Input Fields within the RPC Service
 - Generate an RPC Service (Server Side)
-

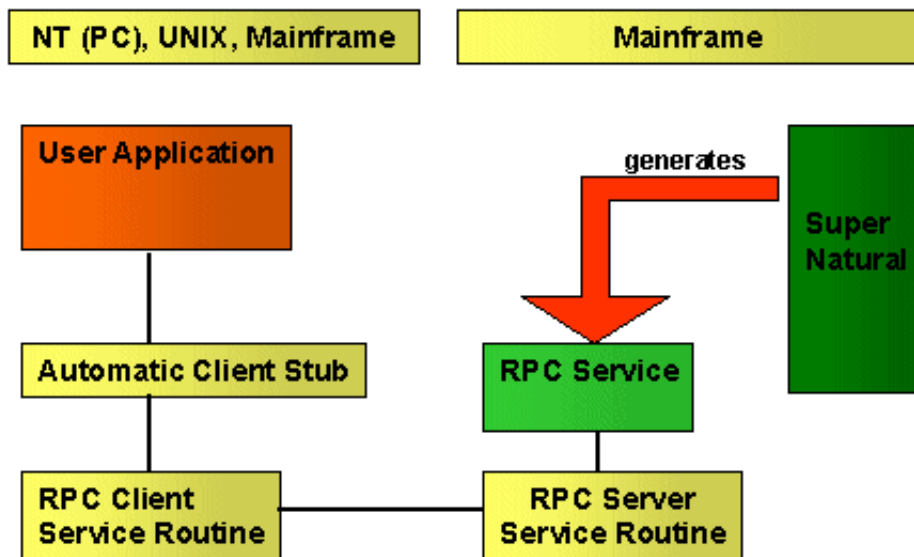
Introduction

Remote procedure call (RPC) techniques establish a framework for communication between server and client systems that can be collocated on the same computer or based on a network of identical or heterogeneous machines and operating systems. The Natural Remote Procedure Call (RPC) facility enables a client Natural program to issue a CALLNAT statement to invoke a subprogram in a server Natural. The Naturalclient and server sessions may run on the same or on a different computer.

Natural RPC exploits the advantages of client server computing. In a typical scenario, Natural on a Windows client computer accesses server data (using a middleware layer) from a Natural on a mainframe computer. The following advantages arise from that:

- The end user on the client can use a Natural application with a graphical user interface.
- A large database can be accessed on a mainframe server.
- Network traffic can be minimized when only relevant data are sent from client to server and back.

The RPC (Remote Procedure Call) Service is designed to produce Super Natural services for distributed working. Super Natural RPC generates a Natural RPC subprogram from a Super Natural Transaction. This NaturalRPC subprogram represents a Natural object that is executable in any Natural RPC environment. A NaturalRPC service (subprogram) that was generated from Super Natural is executable via a Natural RPC server. The RPC client can be in a network or on a local system and is completely independent.



For further information about RPC see *Natural RPC Documentation*.

Hints and Limitations

The following points provide information on the RPC service and show which transactions are not supported:

- Before you can start using the RPC Service the administrator has to set it up in the user profile.
- The RPC Service is generated in Natural Reporting Mode.
- The RPC Service can be used as any Natural RPC Service. It contains no Super Natural-specific information.
- For each RPC Service generation, the transaction is automatically saved with the Super Natural command **KEEP**.
- You can only use Report Transactions to generate the RPC Service, no Data Maintenance Transactions.
- Transactions, defined with report type `Tabular` or `Connect` are not supported by the RPC Services.
- The RPC Service Objects are generated to be used with automatic Natural RPC execution. This requires to set the RPC parameter to `AUTORPC=ON`. For further information see the *Natural Documentation* section **Remote Procedure Call > Automatic Natural RPC Execution**.



The library type is important to meet the requirements of the RPC logon feature that is set on the client side. For further information see *Natural Documentation* section **Remote Procedure Call > Logon Features**.

Dynamic Input Fields

You can process more dynamic Super Natural RPC by using the Super Natural dynamic input fields. Herewith the client application defines the selection values at execution time when calling the RPC Service.

This requires an additional input structure to be generated for the RPC Service. It must be part of the client interface.

Generate an RPC Service

You can use any transaction (except the documented limitations) of your worksheet to proceed the RPC Service.

To generate an RPC Service

1. Select a transaction out of the transaction list with the modify (MO) command.

The worksheet appears.

2. Enter GENERATE RPC into the command line.
3. Press Enter.

The following window with the default settings appears:

RPC-Service Generation

The RPC Service will be stored with
Object Name RPCSMPL1
in Library NSNSAMPL
with Generation Mode .. Direct
and Record Limit 545

☐ Generate with these options
☒ Change options before generation
☐ Modify Transaction

4. Specify how you want to proceed:

Generate with these options

The generation process starts. If the service object to be generated already exists, confirm the replacement in the appearing window.

Change options before generation

Modify the RPC service options before starting the generation process. The Modify Generation Options window appears.

Modify Transaction

Return to the preceding screen to modify the transaction.

5. Select Change Options before Generation.
6. Press Enter.

The following screen appears:

RPC-Service - Modify Generation Options	
RPC Service Object Name	RPCSMPL1
Library	NSNSAMPL
Generation Mode	D Direct
Limit of Result Records	545_____
(Max. possible Result Records .. 545 _____)	

7. Fill in the required information.

RPC Service Object Name

Name of the RPC Service Object to be generated. Depends on the user options and can be pre-defined by the administrator.

Library

Library of the RPC Service Object to be stored. Depends on the user options and can be pre-defined by the administrator.

Generation Mode

Direct or Context. Defines how the Generation will be processed.

Limit of Result Records / Max. possible Result Records

Result record handling. For further information see section Result Data Limitations.

8. Press Enter.

The RPC Service Object has been generated and is ready to use.

You can control your RPC Service by looking at the Natural-Code.



To confirm the replacement of an RPC Service

If you want to generate the same RPC Service object twice you have to confirm the replacement of the first.

1. Specify the RPC Service Object name in the RPC Options window and press Enter.

The following window appears:

```

!-----!
!           Confirm Replace RPC Service           !
! !                                               !
! The RPC Service RPCSMPL1                       !
!   in Library NSNSAMPL                         !
! does already exist.                           !
! The object will be replaced now.               !
! !                                               !
! Enter Y to confirm: Y                         !
!-----!

```

2. Confirm the replacement by entering "Y" and press Enter

Result Data Limitations

Direct Mode

The number and the length of fields are relevant for the number of records!

Example selected result data:

#FIELD - A	N 8
#FIELD - B	A 1
#FIELD - C	A 20
#FIELD - D	A 20

- results in a record length of 49 Byte
- maximal parameter buffer is 30 KByte
- results in a maximal number of 612 records

Context Mode

The largest field length is relevant for the number of records in the context area (intermediate storage). The number of records retrieved with one client call has the same limitation as explained in direct mode but multiple calls can be performed in the context mode.

Example selected result data:

#FIELD - A	N 8
#FIELD - B	A 1
#FIELD - C	A 20
#FIELD - D	A 20

- largest field length is 20 Byte
- maximal array size for a field in the context area is 32 KByte
- results in a maximal number of 1638 records